

VU Research Portal

Evolving robot software and hardware

Eiben, A. E.

published in

SEAMS 2020

2020

DOI (link to publisher)

[10.1145/3387939.3391593](https://doi.org/10.1145/3387939.3391593)

document version

Publisher's PDF, also known as Version of record

document license

Article 25fa Dutch Copyright Act

[Link to publication in VU Research Portal](#)

citation for published version (APA)

Eiben, A. E. (2020). Evolving robot software and hardware. In *SEAMS 2020: Proceedings of the IEEE/ACM 15th International Symposium on Software Engineering for Adaptive and Self-Managing Systems* (pp. 1-4). Association for Computing Machinery, Inc. <https://doi.org/10.1145/3387939.3391593>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:

vuresearchportal.ub@vu.nl

Evolving Robot Software and Hardware

A.E. Eiben

a.e.eiben@vu.nl

Vrije Universiteit Amsterdam



Figure 1: The first robot family created by the Robot Baby Project at the Vrije Universiteit Amsterdam [15]. The parents are the green and the blue robots on the right. The offspring, created through crossover and mutation, is the robot on the left.

ABSTRACT

This paper summarizes the keynote I gave on the SEAMS 2020 conference. Noting the power of natural evolution that makes living systems extremely adaptive, I describe how artificial evolution can be employed to solve design and optimization problems in software. Thereafter, I discuss the Evolution of Things, that is, the possibility of evolving physical artefacts and zoom in on a (r)evolutionary way of creating ‘bodies’ and ‘brains’ of robots for engineering and fundamental research.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; Robotics.

KEYWORDS

Evolutionary Computation, Evolutionary Robotics, Learning

ACM Reference Format:

A.E. Eiben. 2020. Evolving Robot Software and Hardware. In *IEEE/ACM 15th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS '20)*, October 7–8, 2020, Seoul, Republic of Korea. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3387939.3391593>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SEAMS '20, October 7–8, 2020, Seoul, Republic of Korea
© 2020 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-7962-5/20/05.
<https://doi.org/10.1145/3387939.3391593>

1 EVOLUTIONARY ALGORITHMS

Evolution is arguably an excellent designer and optimiser. It has driven the emergence of Life on Earth creating a vast diversity of lifeforms adapted to all kinds of environmental conditions. The ‘engine’ behind evolution is the reproduction-selection cycle that is analogous to the generate-and-test loop of search algorithms. This has been noticed many decades ago [2, 22] and by the end of the twentieth century evolutionary computing (EC) has become a vibrant research area with many applications [8].

The fundamental insight behind EC is to link natural evolution and search-based problem solving, perceive the problem context as an environment, candidate solutions as individuals in this environment, and the quality of a candidate solution as its fitness that determines its chances for survival and reproduction. Based on this perspective, an evolutionary problem solving process can be conducted by creating a population of candidate solutions and consecutively updating this population by (randomized) selection and reproduction such that the candidate solutions with a higher fitness have a higher chance to survive and reproduce. The general evolutionary loop is shown in Figure 2.

An important technical detail under the hood is another bio-inspired ‘trick’: the distinction between the genotype and the phenotype of the individuals. The genotype is the code, the technical analogy of DNA, while the phenotype is the real object in the context of the problem to be solved, the technical analogy of the organism encoded by the given piece of DNA. Crucially, the reproduction operators (crossover and mutation) are applied to the genotypes, while the selection operators (parent selection and survivor selection) work on the phenotypes. A given optimisation

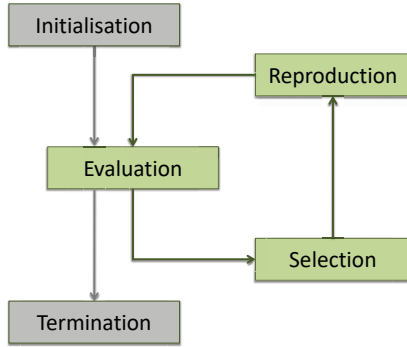


Figure 2: General diagram of Evolutionary Algorithms.

problem naturally determines the phenotypes. For instance, for a travelling salesman problem phenotypes are possible routes. However, the user can freely choose the genotypes to represent these. For the travelling salesman problem, a route can be coded as a list of city names or by an adjacency matrix. The choice of the genotypes and the definition of the corresponding genotype-phenotype mapping (a.k.a. representation) is one of the most important design decisions when specifying an evolutionary algorithm [21].

Over the last decades, EAs have been developed under different names including genetic algorithms, genetic programming, evolution strategies, and they have proven their power on hard problems without analytical models, non-linear relations among variables and complex objective functions with multiple local optima [8, 9].

2 THE EVOLUTION OF THINGS

Evolutionary computing mimics natural evolution, but there are significant differences between them, cf. Table 1 in [9]. Perhaps the most prominent difference is that evolutionary computing is, well, ... computing. That is, it takes place in a virtual space, whereas natural evolution happens in the real world. The advantage of evolutionary computing systems is that they are programmable, configurable, and observable. Natural evolutionary systems are quite the opposite. They are certainly real, but hardly programmable, configurable, and observable. The combination of the two offering the best of both worlds is the *The Evolution of Things* as introduced in [7] and further discussed in [10] and [9], cf. Figure 3.

The key idea behind the Evolution of Things concept is to have a programmable evolutionary system that works with physical artefacts. These artefacts can be passive, e.g., sunglasses or airplane wings, or active, animate things, robots for short. Robots that are able to reproduce and evolve in the real world hold great promises for engineering –these will be discussed in the next section– as well as for fundamental research. To this end, an evolving robot system can be perceived as a hardware model of natural evolution [19] and used as a research instrument to study evolutionary phenomena. Fundamental questions that can be investigated include the evolution of (embodied) intelligence, the interplay between the body and the brain, and the impact of the environment on the evolved organisms. Using real robots instead of simulations is interesting, because this guarantees that the observed effects are real and not just artefacts of the simulator. Research with robots also

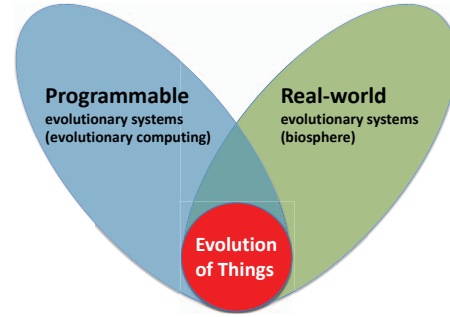


Figure 3: Evolution of Things: the best of both worlds.

offers advantages with respect to living organisms, because robots are easily observable (e.g., internal processes and communication can be logged) and controllable which allows systematic studies under strictly regulated conditions and many repetitions for solid statistics.

3 EVOLUTIONARY ROBOTICS

Evolutionary Robotics (ER) is a research area that applies EAs to design and optimize the bodies (morphology, hardware), the brains (controller, software) or both for simulated or real autonomous robots [4, 20, 23]. Using artificial evolution for robot design has a strong rationale.

As natural evolution has produced successful life forms for practically all possible environmental niches on Earth, it is plausible that artificial evolution can produce specialised robots for various environments and tasks.

Obviously, designing robots for structured environments with known and predictable conditions can be done by classic engineering. However, complex unstructured environments with (partially) unknown and possibly changing conditions represent a completely different challenge. Think, for instance, of robots for environmental monitoring in rain forests, exploration of ocean floors, or terraforming on other planets. In such cases it is hard to determine the optimal morphologies and the control systems driving them. For example, should a robot that operates in the jungle have wheels, legs, or both? What is the optimal arrangement of its sensors? Should that robot be small to maneuver through narrow openings or should it be big and heavy to trample down obstacles?

Evolutionary algorithms have been successful in solving various design problems and obtaining unexpected, ‘original’ solutions that surprised their users [1, 12, 17]. To illustrate their potential for designing robots let us consider two motivational examples.

Example 1: Breeding Farm Recall the problem of designing robots for inspecting forests. An evolutionary approach to this problem can be implemented on a ‘robot breeding farm’ that consists of a mock-up forest environment and an evolutionary engine. Reproduction could be realized by a rapid prototyping facility that constructs

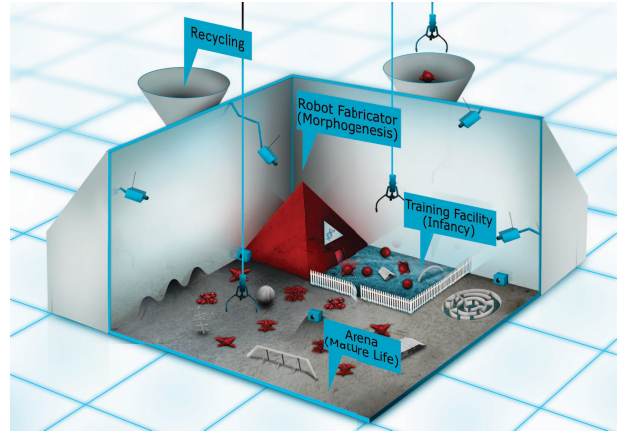
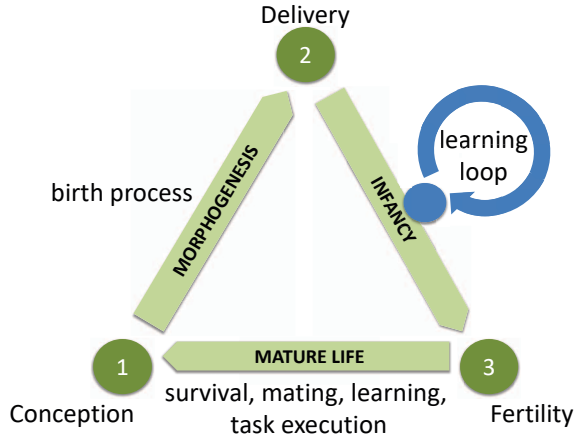


Figure 4: Left: Generic system architecture for robot evolution conceptualized by the Triangle of Life, cf. [6]. Right: Illustration of an EvoSphere that stands for a tangible implementation of this framework [5]. An EvoSphere consists of three main components that belong to the three edges of the triangle, plus a recycling facility and equipment for observation.

the robots (phenotypes) based on specs sheets (genotypes). Fitness evaluations can be done by measuring the task performance of the robots in the test environment and selection can be implemented by using this notion of fitness. Additionally, the users can steer and accelerate evolution by (de)selecting robots for reproduction as they see fit, acting akin to farmers who breed animals. Robot breeders can stop the evolutionary process after obtaining a good robot and produce several copies of it to be deployed in the real-world.

In the Breeding Farm example, and in most evolutionary computing applications, evolution is (ab)used as an optimizer that is halted when a satisfactory solution is found. Real evolution, however, is not about optimization, but about adaptation that never stops. Including this feature in robot populations would imply that they can adapt to previously unknown and/or changing conditions.

Example 2: Terraforming Imagine a mission for the colonization of a moon or another planet. Using robots to explore the planet and make it habitable is a straightforward option, but designing an optimal morphology and control system in advance can be unfeasible. An evolutionary engine operating autonomously on the planet can mitigate this problem. The first component of this system is a (re)production facility that can make use of local resources and construct a large variety of robots. The second one is a twofold selection drive, such that robots become fit for the environment as well as fit for purpose. Environmental selection (for viability) is for free, as robots with a poor feature set will not be able to operate adequately. Sexual selection, in turn, can be pre-programmed such that robots have a ‘basic instinct’ to choose mating partners with a high task performance (utility). The evolving robot population will then become increasingly adapted to the given planet and adjust their bodies and brains when the conditions change - something that is essential in terraforming.

Note, that the case of terraforming is very different from the breeding farm because the evolutionary system must operate for

extended periods of time without direct human oversight. The evolutionary process should keep running such that the robots can continually adapt their bodies and brains over consecutive generations. In this respect the terraforming application is closer to biological evolution, while a breeding farm is more like a usual evolutionary design process.

These examples, although quite different, share an important element: both describe a system of real robots, where the morphologies as well as the controllers undergo evolution. This is in stark contrast with the current practice. Evolutionary Robotics today is mainly concerned with evolving the controllers of simulated robots. ER systems where morphologies and controllers of robots evolve simultaneously are rare and –forced by technological limitations– they only work in simulation. Occasionally, an evolved robot is constructed in the real world, but the evolutionary process is simulated [18]. This practice inevitably runs into the reality gap [14], that is, the phenomenon that a solution evolved in simulation will only work well in simulation. Implementing the evolved solution in the real world, on a physical robot, typically leads to a different behaviour that is (much) worse than the simulated one.

4 HOW TO EVOLVE REAL ROBOTS

The 2012 position paper [7] outlined the promises and grand challenges of artificial evolutionary processes *in materio*, but it did not provide implementation guidelines. These have been offered in [6] that presented an framework for “evolving robots in real time and real space”. A tangible implementation of this framework is envisaged by the notion of an EvoSphere as discussed in [5].

This framework, called the Triangle of Life, is illustrated in Figure 4. It consists of three stages: morphogenesis, infancy, and mature life, consequently, an EvoSphere consists of three main components: the Robot Fabricator, the Training Facility, and the Arena. The Robot Fabricator is where new robots are created. The Training Facility hosts a learning environment for ‘infant’ robots so they can learn

to control their –possibly unique– body to acquire basic skills (locomotion, object manipulation) and to perform more complex tasks. If a robot achieves a satisfactory level of performance, it is declared a fertile adult and enters the Arena which represents the world where the robots must survive and perform user-defined tasks. Successful robots can be selected for reproduction. The sexual selection mechanism can be innate in the robots so they choose ‘mating partners’ autonomously, or executed by an overseer, which can be algorithmic, a human breeder, or a combination.

Because a robot is a combination of its body (morphology, hardware) and its brain (controller, software), it is necessary that the genotypes code for both the body and the brain. Crossover and mutation act on the genotypes of the parents and produce a new genotype, where the body coding segment is the combination of the body coding segments of the parents. The body of the offspring is then created by the Robot Fabricator. The evolution of robot brains is partly similar, the offspring starts its life cycle with the combination of the brain coding segments of the parents. However, this inherited brain undergoes a learning process in the Training Facility before the robot can enter the Arena.

Including a learning facility is not an arbitrary design choice, it is meant to mitigate a general problem. Namely, while it can be assumed that the parents had well-matching bodies and brains (otherwise they had not been fit enough to be selected for mating), in general it cannot be assumed that crossover preserves the good match. Thus, we must cope with a potential mismatch between the inherited body and the inherited brain of the offspring. This implies that the evolution of brains is different from the evolution of bodies. Bodies undergo ‘clean’ evolution, whereas the adaptation of brains is driven by a combination of evolution and learning. Interestingly, the learning method can be implemented by an EA (e.g., neuro-evolution, genetic programming, evolution strategy). In this case, we get a system with two evolutionary loops. The outer loop that forms the Triangle of Life is evolving bodies and brains, while the inner loop under the hood of the learning method is improving the brain in the given body of a newborn robot.

5 CHALLENGES AHEAD

With the development of 3D-printing, rapid prototyping, and automated assembly the evolution of robots is becoming feasible, at least in an academic setting [3, 13, 15, 24]. Additionally, researchers have demonstrated how an organism evolved in simulation can be incarnated by using ‘wetware’ instead of hardware [16]. Recently, the first large scale research project, *Autonomous Robot Evolution: Cradle to Grave*¹ has been commenced [11]. The ARE project represents a big step towards real applications and it illuminates the most critical challenges: 1) Automated construction of fully functional robots; 2) Time and sample efficient learning of multiple skills and tasks; 3) Reliable evaluation of task performance and fitness. Meeting these challenges will help unlock the full potential of robot evolution for engineering as well as for fundamental research. Over the long term, this will lead to a new breed of machines that can change their form and behaviour, not by error, but on purpose.

ACKNOWLEDGMENTS

I gratefully acknowledge the financial support of the Vrije Universiteit Amsterdam and the EPSRC under the ARE project: EP/R03561X, EP/R035679, EP/R035733.

REFERENCES

- [1] P.J. Bentley (Ed.). 1999. *Evolutionary Design by Computers*. Morgan Kaufmann.
- [2] H.J. Bremermann, M. Rogson, and S. Salaff. 1966. Global Properties of Evolution Processes. In *Natural Automata and Useful Simulations*, H.H. Pattee, E.A. Edsack, L. Fein, and A.B. Callahan (Eds.). Spartan Books, Washington DC, 3–41.
- [3] Luzius Brodbeck, Simon Hauser, and Fumiya Iida. 2015. Morphological evolution of physical robots through model-free phenotype development. *PLoS one* 10, 6 (2015), e0128444.
- [4] S. Doncieux, N. Bredeche, J.-B. Mouret, and A.E. Eiben. 2015. Evolutionary robotics: what, why, and where to. *Frontiers in Robotics and AI* 2 (2015), 4.
- [5] A.E. Eiben. 2015. EvoSphere: The World of Robot Evolution. In *Proc. of TPNC 2015 (LNCS 9477)*, A.-H. Dediu, L. Magdalena, and C. Martin-Vide (Eds.). Springer, 3–19. https://doi.org/10.1007/978-3-319-26841-5_1
- [6] A.E. Eiben, N. Bredeche, M. Hoogendoorn, J. Stradner, J. Timmis, A.M. Tyrrell, and A. Winfield. 2013. The Triangle of Life: Evolving robots in real-time and real-space. In *Proc. of ECAL 2013*, P. Lio, O. Miglino, G. Nicosia, S. Nolfi, and M. Pavone (Eds.). MIT Press, 1056–1063.
- [7] A.E. Eiben, S. Kernbach, and E. Haasdijk. 2012. Embodied Artificial Evolution – Artificial Evolutionary Systems in the 21st Century. *Evolutionary Intelligence* 5, 4 (2012), 261–272.
- [8] A.E. Eiben and J.E. Smith. 2003. *Introduction to Evolutionary Computing*. Springer, Berlin Heidelberg.
- [9] A.E. Eiben and J.E. Smith. 2015. From evolutionary computation to the evolution of things. *Nature* 521, 7553 (2015), 476.
- [10] A. E. Eiben. 2014. In Vivo Veritas: towards the Evolution of Things. In *Proc. of PPSN XIII (LNCS 8672)*, T. Bartz-Beielstein, J. Branke, B. Filipič, and J. Smith (Eds.). Springer, 24–39.
- [11] Matthew F Hale, Edgar Buchanan, Alan F Winfield, Jon Timmis, Emma Hart, Agoston E Eiben, Mike Angus, Frank Veenstra, Wei Li, Robert Woolley, et al. 2019. The ARE Robot Fabricator: How to (Re) produce Robots that Can Evolve in the Real World. In *Proc. of ALIFE 2019*. MIT Press, 95–102.
- [12] Gregory. S. Hornby, Jason D. Lohn, and Derek S. Linden. 2011. Computer-Automated Evolution of an X-Band Antenna for NASA’s Space Technology 5 Mission. *Evolutionary Computation* 19, 1 (2011), 1–23.
- [13] David Howard, Agoston E Eiben, Danielle Frances Kennedy, Jean-Baptiste Mouret, Philip Valencia, and Dave Winkler. 2019. Evolving embodied intelligence from materials to machines. *Nature Machine Intelligence* 1, 1 (2019), 12–19.
- [14] Nick Jakobi, Phil Husbands, and Inman Harvey. 1995. Noise and the reality gap: The use of simulation in evolutionary robotics. In *European Conference on Artificial Life*. Springer, 704–720.
- [15] Milan Jelisavcic, Matteo De Carlo, Elte Hupkes, Panagiotis Eustratiadis, Jakub Orłowski, Evert Haasdijk, Joshua E Auerbach, and Agoston E Eiben. 2017. Real-world evolution of robot morphologies: A proof of concept. *Artificial life* 23, 2 (2017), 206–235.
- [16] Sam Kriegman, Douglas Blackiston, Michael Levin, and Josh Bongard. 2020. A scalable pipeline for designing reconfigurable organisms. *Proceedings of the National Academy of Sciences* 117, 4 (2020), 1853–1859. <https://doi.org/10.1073/pnas.1910837117> arXiv:<https://www.pnas.org/content/117/4/1853.full.pdf>
- [17] Joel Lehman, Jeff Clune, and Dusan Misevic. 2018. The Surprising Creativity of Digital Evolution. In *Proc. of ALIFE 2018*. MIT Press, 55–56.
- [18] Hod Lipson and Jordan B Pollack. 2000. Automatic design and manufacture of robotic lifeforms. *Nature* 406, 6799 (2000), 974.
- [19] John Long. 2012. *Darwin’s Devices: What Evolving Robots Can Teach Us About the History of Life and the Future of Technology*. Basic Books.
- [20] Stefano Nolfi and Dario Floreano. 2000. *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*. MIT press.
- [21] Franz Rothlauf. 2006. *Representations for genetic and evolutionary algorithms* (second ed.). Springer-Verlag.
- [22] A.M. Turing. 1948. Intelligent Machines. In *Mechanical Intelligence: Collected Works of A.M. Turing*, D.C. Ince (Ed.). North-Holland, Amsterdam, 107–128.
- [23] Patricia A. Vargas, Ezequiel A. Di Paolo, Inman Harvey, and Phil Husbands. 2014. *The Horizons of Evolutionary Robotics*. The MIT Press.
- [24] Vuk Vujovic, Andre Rosendo, Luzius Brodbeck, and Fumiya Iida. 2017. Evolutionary Developmental Robotics: Improving Morphology and Control of Physical Robots. *Artificial Life* 23, 2 (2017), 169–185.

¹<https://www.york.ac.uk/robot-lab/are/>